# 一种基于信息熵的混合属性数据谱聚类算法 *

姜智涵[1,2,3]，朱 军[1,3]，周晓锋[1,3]，李 帅[1,2,3]

(1. 中国科学院沈阳自动化研究所，沈阳 110016; 2. 中国科学院大学，北京 100049; 3. 中国科学院网络化控制系统重点实验室，沈阳 110016)

**摘 要：**针对传统的聚类算法只能处理单属性的数据，不能很好地处理混合属性数据的聚类问题，以及目前大多数混合属性数据聚类算法对初始化敏感、不能处理任意形状的数据的问题，提出一种基于信息熵的混合属性数据谱聚类算法，用于处理混合类型数据。首先，提出了一种新的相似性度量方式，利用谱聚类算法中的数值型数据构成的高斯核函数矩阵与新的基于信息熵的分类型数据构成的影响因子矩阵相结合代替了传统的相似度矩阵，新的相似度矩阵避免了数值属性与分类属性数据之间的转换和参数调整；然后，把新的相似度矩阵运用到谱聚类算法中，以便于处理任意形状的数据，最终得出聚类结果。通过在 UCI 的数据集上的实验表明，该算法能有效地处理混合属性数据的聚类问题，且具有较高的稳定性以及良好的鲁棒性。

**关键词：**混合属性数据；谱聚类；高斯核函数；影响因子

**中图分类号：**TP doi: 10.3969/j.issn.1001-3695.2018.02.0080

# Entropy-based spectral clustering algorithm for mixed type data

Jiang Zhihan[1,2,3], Zhu Jun[1,3], Zhou Xiaofeng[1,3], Li Shuai[1,2,3]

(1. *Shenyang Institute of Automation*, *Chinese Academy of Sciences*, *Shenyang* 110016, *China*; 2. *University of Chinese Academy of Sciences*, *Beijing* 100049, *China*; 3. *Key Laboratory of Network Control System*, *Chinese Academy of Sciences*, *Shenyang* 110016, *China*)

**Abstract:** Aiming at the problem that the traditional clustering algorithm can only deal with single attribute data and can't handle the clustering problem of mixed type data very well. Most of the clustering algorithms for mixed type data currently have the problem of initializing sensitive and can't handle the data of arbitrary shape. This paper proposed an entropy-based spectral clustering algorithm for mixed type data to deal with mixed type data. First, it proposed a new similarity measure. It used the numerical data in the spectral clustering algorithm constitutes a Gaussian kernel function of the matrix, and used the classification data constitutes an entropy-based the influence factor of the matrix. A new similarity matrix combines these two matrices. Instead of the traditional similarity matrix, it proposed the new similarity matrix avoid feature transformation and parameter adjustment between the numerical data and the classification data. Then, it applied the new similarity matrix to the spectral clustering algorithm so as to deal with the data of arbitrary shape, and finally got the clustering result. Experiments on UCI data sets show that this algorithm can effectively deal with the clustering problem of mixed attribute data, with high stability and good robustness.

**Key words**: mixed type data; spectral clustering; Gaussian kernel function; influence factor

## 0 引言

聚类分析的目的是在数据集的子集之间寻找相关性，并评估这些子集中的元素之间的相似性[1,2]。聚类在包括生物学、经济学和医学在内的各个领域都有很多应用。它的应用包括数据挖掘、文档检索、图像分割和模式识别[3]。传统的聚类方法只能处理单属性的数据，如 K-means[4]、RDBSCAN [5]、CSSA-OIK[6]、基于竞争思想的分级聚类[7]等算法只针对处理数值型数据，而 K-modes [8]、COOLCAT [9]、CECD[10]等算法只针对处理分类型数据。在处理混合属性数据时，上述的算法都得不到期望的聚类效果[11]。

处理混合类型数据的一种直接处理方式是将分类属性转换

---

**收稿日期：**2018-02-01；**修回日期：**2018-03-24 **基金项目：**工信部智能制造综合标准化与新模式应用项目（Y6L8283A01）

**作者简介：**姜智涵（1992-），男，山东烟台人，硕士研究生，主要研究方向为机器学习、大数据 (jiangzhihan@sia.cn)；朱军（1964-），男，研究员，硕士，主要研究方向为大型分布控制系统集成、智能控制技术及应用、网络自动控制系统的研究和开发；周晓锋（1978-），女，副研究员，博士，主要研究方向为数据挖掘、机器学习；李帅（1988-），男，博士研究生，助理研究员，主要研究方向为数据挖掘、机器学习、过程监测、故障诊断.

为新的形式，如二进制字符串，然后应用到前面提到的基于数值属性的聚类算法中。但是二进制编码有三个缺点：首先，这种方法破坏了分类属性的原始结构。为了避免分类属性之间的参数调整，Li 等人[12]提出了一个基于混合数据相似度度量的 SBAC 算法。其次，如果分类属性的定义域很大，那么转换后的二进制就会有更大的维度。最后，维护的难度。如果将属性值添加到分类属性中，那么所有对象将被更改。为了更好地解决这个问题，许多研究人员在过去十几年里，基于相似度指标直接研究了分类属性。一些方法基于相似度的度量指标考虑了数值属性和分类属性，如 K-prototypes [13]。然而考虑到数据在簇归属上的不确定性，Chatzis 等人 [14]提出了 KL-FCM-GM 算法来扩展 K-prototypes 算法，KL-FCM-GM 算法是假设簇中的数据符合高斯分布。还有一些是基于无参数的相似度度量的方法，如 OCIL [15].但是这种度量方式只能度量一个对象与一个簇之间的相似性。就像 K-prototypes 算法一样，OCIL 使用 K-means 的形式来对混合类型数据进行聚类，是一种迭代的聚类算法。因此，这种算法对初始化很敏感，适用于球面分布的数据[16]。

针对传统的聚类算法只能处理单属性的数据，不能很好地处理混合属性数据的聚类问题，以及目前大多数混合属性数据聚类算法对初始化敏感、不能处理任意形状的数据的问题，本文提出了一种基于信息熵的混合属性数据谱聚类(EBSCMD)算法。该算法利用谱聚类算法中的数值型数据构成的高斯核函数矩阵与新的基于信息熵的分类型数据构成的影响因子矩阵相结合代替了传统的相似度矩阵，避免了数值属性和分类属性数据之间的转换和参数调整，再把新的相似度矩阵运用到谱聚类算法中处理任意形状的数据，最终得出聚类结果。为了验证 EBSCMD 算法的可行性以及有效性，本文利用一些 UCI 数据集进行了一些实验，并与其他算法进行了比较。实验结果表明，EBSCMD 算法能有效地处理混合属性数据的聚类问题，且具有较高的稳定性及鲁棒性。

## 1 谱聚类算法及其相关定义

谱聚类算法是一种基于谱图理论的聚类算法，其本质是将聚类问题转换为图的最优划分问题[17]。与传统的聚类算法相比，谱聚类算法具有能在任意形状的样本空间上聚类且收敛于全局最优解的优点。其主要思想是把所有的数据对象看做空间中的点，这些点之间可以用边连接起来。距离较远的两点之间的边权重值较低，而距离较近的两点之间的边权重值较高。通过对所有数据点组成的图进行切图，让切图后不同的子图间的边权重和尽可能的低，而子图内的边权重和尽可能的高，从而达到聚类的目的。在构造适当的图基础上，将原来的聚类问题转换为图论中的子图最优划分问题[18]。

谱聚类算法一般分为三步[19]：

a) 根据数据之间的相似度建立相似度矩阵；

b) 求相似度矩阵最小的 k 个特征值对应的特征向量，并构成新的特征向量空间；

c) 使用 K-means 算法对这个新的向量空间聚类，输出聚类结果。

算法中相关定义如下：

**定义 1** 无向权重图。对于一个图 ***G***，一般用点的集合 ***V*** 和边的集合 ***L*** 来描述，即为 ***G(V,L)***。其中 ***V*** 即为数据集里面所有的点($v_1$，$v_2$，…$v_n$)。对于 ***V*** 中的任意两点，可以有边连接，也可以没有边连接。定义权重$w_{ij}$为点$v_i$与点$v_j$之间的权重。由于是无向图，所以$w_{ij} = w_{ji}$。

**定义 2** 度矩阵 ***D***。对于有边连接的两个点$v_i$和$v_j$，$w_{ij}>0$，对于没有边连接的两个点$v_i$和$v_j$，$w_{ij}=0$。对于图的任意一个点$v_i$，它的度$d_i$定义为它相连的所有边的权重之和，即

$$d_i = \sum_{j=1}^{n} w_{ij} \tag{1}$$

利用每个点度的定义，本文可以得到一个 $n\times n$ 的度矩阵 ***D***。它是一个对角矩阵，对应第 i 行的第 i 个点的度数，定义如下：

$$\boldsymbol{D} = \begin{bmatrix} \boldsymbol{d}_1 & 0 & \cdots & 0 \\ 0 & \ddots & \cdots & \vdots \\ \vdots & \cdots & \ddots & 0 \\ 0 & \cdots & \cdots & \boldsymbol{d}_n \end{bmatrix} \tag{2}$$

**定义 3** 邻接矩阵 ***W***。利用所有点之间的权重值，可以得到图的邻接矩阵 ***W***。它也是一个 $n\times n$ 的矩阵，第 i 行的第 j 个值对应权重$w_{ij}$。这里定义邻接矩阵的方法是全连接法，此方法的相似矩阵与邻接矩阵相同。$w_{ij}$计算采用高斯核函数，即

$$w_{ij} = \exp\left(-\frac{\| x_i - x_j \|_2^2}{2\sigma^2}\right) \tag{3}$$

**定义 4** 拉普拉斯矩阵 ***L***。拉普拉斯矩阵是对称矩阵，由 ***D*** 和 ***W*** 相减得到的，即 ***L=D-W***。

**定义 5** 相似矩阵 S。由样本点距离度量组成的矩阵。

对于数值型数据，一般利用欧氏距离作为数据之间的相似性度量，然后利用高斯核函数把其转换为无向加权图边上的权值，构建出一个关于数值型数据的高斯核函数矩阵。用这种方法处理能很好地解决数值型数据聚类问题，并能取得全局最优解。对于分类型数据，若利用欧氏距离表示混合数据之间的相似性度量，把分类型数据转换为数字，然后计算欧氏距离。显然不能准确反映数据的内在联系，所以利用谱聚类算法解决混合数据聚类问题的关键在于为混合数据定义一个合适的相似性度量，并为之选择合适的相似度矩阵。

## 2 基于信息熵的混合属性数据谱聚类算法

本文提出了一个新的相似性度量作为一个统一的框架用于处理包含数值型数据和分类型数据的混合型数据。为了更好地对任意形状的数据进行聚类，将新的相似性度量方式引入到谱

聚类算法中。同时，为了更好地适应谱聚类算法的算法流程，本文构建了一个新的分类型数据的相似度矩阵与高斯核函数矩阵相融合运用到谱聚类算法中。

## 2.1 相似性度量矩阵

关于混合属性数据的相关公式符号描述如下：$\boldsymbol{X}=\{x_1,x_2,\cdots,x_n\}$表示N个混合数据对象的数据集，对于每一个$x_i$，$i\in[1,\mathrm{N}]$，混合型数据集$x_i$代表M($M=M_n+M_c$)个属性$A_1^{(n)},A_2^{(n)},\cdots,A_{M_n}^{(n)},A_{M_n+1}^{(n)},\cdots,A_{M_n+M_c}^{(c)}$,这里$A_1^{(n)},A_2^{(n)},\cdots,A_{M_n}^{(n)}$代表$M_n$个数值型数据，$A_1^{(c)},A_2^{(c)},\cdots,A_{M_c}^{(c)}$代表$M_c$个分类型数据。$x_{i,k}^{(n)}$代表$x_i^{(n)}$的第k个属性，$x_i^{(n)}$表示数值部分。$x_{i,k}^{(c)}$代表$x_i^{(c)}$的第k个属性，$x_i^{(c)}$表示分类部分。DOM$\left(A_k^c\right)$（$1\le k\le M_n$）表示$x_{i,k}^{(c)}$的定义域。这个分类属性的定义域表示为DOM$\left(A_k^c\right)=\{a_{k,1},a_{k,2},\cdots,a_{k,r_k}\}$，$r_k$表示第k列分类属性的数量。同时，$x_i$表示一个向量$\left[x_i^{(n)},x_i^{(c)}\right]=\left[x_{i,1}^{(n)},x_{i,2}^{(n)},\cdots,x_{i,\mathrm{M}_n}^{(n)},x_{i,\mathrm{M}_n+1}^{(c)},\cdots,x_{i,\mathrm{M}}^{(c)}\right]$。

与传统的相似性测量方法不同，在用谱聚类算法处理混合属性数据时，本文需要构建一个关于混合属性数据的相似性度量矩阵，这个相似性度量矩阵需要融合数值型数据的相似度矩阵和分类型数据的相似度矩阵。

### 2.1.1 数值型数据的相似度矩阵

对于数值型数据，本文采用谱聚类算法中的高斯核核函数定义$x_i^{(n)}$与$x_j^{(n)}$之间的相似性度量为$\boldsymbol{S_n}\left(x_i^{(n)},x_j^{(n)}\right)$，公式如下：

$$\mathrm{S}_n\left(x_i^{(n)},x_j^{(n)}\right)=\exp\left(-\frac{\|\mathrm{x}_i^{(n)}-\mathrm{x}_j^{(n)}\|_2^2}{2\sigma^2}\right)\tag{4}$$

其中：$\sigma$是一个可调节的参数。

谱聚类算法利用高斯核函数把数据点之间的相似性度量转换为无向加权图边上的权值，构建出一个关于数值型数据的高斯核函数矩阵$\boldsymbol{W}$，

$$\boldsymbol{W}=\begin{bmatrix}0 & \cdots & x_{1j}^{(n)} & \cdots & x_{1n}^{(n)}\\ \vdots & \ddots & \vdots & \cdots & \vdots\\ \vdots & \cdots & x_{ij}^{(n)} & \cdots & \vdots\\ \vdots & \cdots & \vdots & \ddots & \vdots\\ x_{n1}^{(n)} & \cdots & x_{nj}^{(n)} & \cdots & 0\end{bmatrix}\tag{5}$$

$\boldsymbol{W}$是一个$n\times n$（n=N）的矩阵，对角线上的元素全是0，$x_{ij}^{(n)}$代表$x_i^{(n)}$与$x_j^{(n)}$之间的相似性度量$\boldsymbol{S_n}\left(x_i^{(n)},x_j^{(n)}\right)$。

### 2.1.2 分类型数据的相似度矩阵

首先，本文定义$x_{i,k}^{(c)}$与$x_{j,k}^{(c)}$之间的相似性度量为$\boldsymbol{S}_c'\left(x_{i,k}^{(c)},x_{j,k}^{(c)}\right)$。

$$\boldsymbol{S}_c'\left(x_{i,k}^{(c)},x_{j,k}^{(c)}\right)=\begin{cases}1, \text{if } x_{i,k}^{(c)}=x_{j,k}^{(c)}\\0, \text{if } x_{i,k}^{(c)}\ne x_{j,k}^{(c)}\end{cases}\tag{6}$$

这种定义分类型数据的相似性度量是假设每个分类属性的权重是相同的。然而在实践中每个分类属性在分类型数据部分对相似度的计算有不同的贡献，其中一个主要的原因是不同的属性有不同的分布。因此公式可以进一步修改，如下所示：

$$\boldsymbol{S}_c\left(x_{i,k}^{(c)},x_{j,k}^{(c)}\right)=\sum_{k=1}^{M_c}w_k\boldsymbol{S}_c'\left(x_{i,k}^{(c)},x_{j,k}^{(c)}\right)\tag{7}$$

这里$0\le w_k\le1$，$\sum_{k=1}^{M_c}w_k=1$。显然，$w_i$是分类属性$A_k^{(c)}$的权重，它代表了对分类属性部分重要性的计算。

然后，讨论如何计算每个分类属性$A_k^{(c)}$的权重$w_k$。本文把信息熵的概念应用到权重的计算上。数据集中分类型数据的不均匀性越大，分类型数据的信息熵就越大。另外，数据集中的分类属性的不均匀性与分类属性的重要性相对应。因此，根据信息熵公式可以计算分类属性$A_k^{(c)}$，DOM$\left(A_k^c\right)=\{a_{k,1},a_{k,2},\cdots,a_{k,r_k}\}$，定义如下：

$$H_{A_k^{(c)}}=-\sum_{a_{k,t}\in DOM\left(A_k^{(c)}\right)}p\left(a_{k,t}\right)\log_2\left(p\left(a_{k,t}\right)\right),\tag{8}$$

这里$p\left(a_{k,t}\right)$是属性值$a_{k,t}$的概率；$p\left(a_{k,t}\right)=\frac{\sum_{i=1}^{N}S_c'\left(x_{i,k}^{(c)},a_{k,t}\right)}{N}$；分子表示$A_k^{(c)}$中的分类属性值与$a_{k,t}$相等的个数；N是数据集中的对象总数。观察式（8）可以注意到，如果$A_k^{(c)}$中值的数量$r_k$非常大，那么分类属性的信息熵$H_{A_k^{(c)}}$也会很高。这与实际情况是不一样的。为了降低太多不同值甚至唯一值对分类属性的影响，本文重新定义了分类属性的信息熵$H_{A_k^{(c)}}'$：

$$H_{A_k^{(c)}}'=-\frac{1}{r_k}\sum_{t=1}^{r_k}p\left(a_{k,t}\right)\log_2\left(p\left(a_{k,t}\right)\right)\tag{9}$$

因此，可以量化分类属性$A_k^{(c)}$的重要性为

$$w_k=\frac{H_{A_k^{(c)}}'}{\sum_{k=1}^{M_c}H_{A_k^{(c)}}'}\tag{10}$$

将式（10）带入到式（7）中，最终得到分类属性的相似性度量$\boldsymbol{S}_c\left(x_i^{(c)},x_j^{(c)}\right)$，公式如下：

$$\boldsymbol{S}_c\left(x_i^{(c)},x_j^{(c)}\right)=\sum_{k=1}^{M_c}\left(\frac{H_{A_k^{(c)}}'}{\sum_{k=1}^{M_c}H_{A_k^{(c)}}'}.S_c'\left(x_{i,k}^{(c)},x_{j,k}^{(c)}\right)\right)\tag{11}$$

最后，为了更好地适应谱聚类算法的算法流程，需要构建一个分类型数据的相似度矩阵，本文称之影响因子矩阵$\boldsymbol{F}$，形式如下：

$$F=\begin{bmatrix} 0 & \cdots & x_{1j}^{(c)} & \cdots & x_{1n}^{(c)} \\ \vdots & \ddots & \vdots & \cdots & \vdots \\ \vdots & \cdots & x_{ij}^{(c)} & \cdots & \vdots \\ \vdots & \cdots & \vdots & \ddots & \vdots \\ x_{n1}^{(c)} & \cdots & x_{nj}^{(c)} & \cdots & 0 \end{bmatrix} \tag{12}$$

显然，$\boldsymbol{F}$ 同样是一个 $n\times n$（n=N）的矩阵，对角线上的元素全是 0，$x_{ij}^{(c)}$ 代表 $x_i^{(c)}$ 与 $x_j^{(c)}$ 之间的相似性度量 $\boldsymbol{S}_c\left(x_i^{(c)},x_j^{(c)}\right)$。

### 2.1.3 混合型数据的相似度矩阵

从上面的内容可以很容易地发现数值型数据的相似性度量方法是通过高斯核函数把数值型数据点之间的相似性度量转换为无向加权图边上的权值，构建出一个高斯核函数矩阵 $\boldsymbol{W}$。而对于分类型数据来说，分类型数据的相似性度量方法则是利用了信息熵的概念计算每个分类属性的权重，然后利用计算出来的权重把分类型数据点之间的相似性度量乘以对应权重求和，构建了一个影响因子矩阵 $\boldsymbol{F}$。

对于分类属性，本文在建立相似性度量矩阵时，虽然每个分类属性在分类型数据部分对相似度的计算有不同的贡献，但是对分类属性的利用信息熵进行了处理，则可以认为当分类型数据点之间相同的个数越多时，数据间相似性越大；分类属性相同的个数越少时，数据相似性越小。所以本文的混合型数据的相似度矩阵 $\boldsymbol{S}$ 用高斯核函数矩阵 $\boldsymbol{W}$ 和影响因子矩阵 $\boldsymbol{F}$ 点乘得到，公式如下：

$$S=F\cdot W=\begin{bmatrix} S(1,1) & \cdots & \cdots & S(1,n) \\ \vdots & \ddots & \cdots & \vdots \\ \vdots & S(i,j) & \ddots & \vdots \\ S(n,1) & \cdots & \cdots & S(n,n) \end{bmatrix} \tag{13}$$

## 2.2 算法实现

### 2.2.1 EBSCMD 算法的实现流程

EBSCMD 算法流程如图 1 所示。

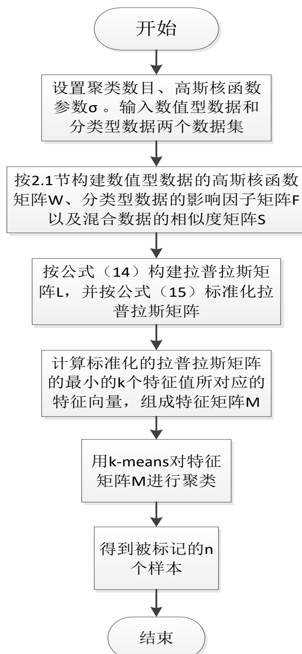



图 1 EBSCMD 算法流程

按 EBSCMD 算法过程描述如下：

输入：混合型数据中的数值型数据和分类型数据两个数据集，聚类个数 k。

输出：被标记聚类类别的 n 个样本，即 $C=(c_1,c_2,\cdots,c_n)$。

a)对输入的数值型数据进行标准化处理，构建数值型数据的高斯核函数矩阵 $\boldsymbol{W}$。

b)对输入的分类型数据利用信息熵公式计算每个分类属性的权重，构建分类型数据的影响因子矩阵 $\boldsymbol{F}$。高斯核函数矩阵 $\boldsymbol{W}$ 与影响因子矩阵 $\boldsymbol{F}$ 点乘得到混合数据的相似度矩阵 $\boldsymbol{S}$。

c)根据相似度矩阵 $\boldsymbol{S}$ 得到度矩阵 $\boldsymbol{D}$，构建拉普拉斯矩阵 $L$，公式如下：

$$L=D-S \tag{14}$$

标准化拉普拉斯矩阵得到 $L^{'}$，公式如下：

$$L^{'}=D^{-1/2}LD^{-1/2} \tag{15}$$

其中：度矩阵 $\boldsymbol{D}$ 对角元素 $d_i=\sum_{j=1}^{n}S(i,j)$。

d)计算标准化的拉普拉斯矩阵 $L^{'}$ 的最小的 $k_1$ 个特征值所对应的特征向量，并对 $k_1$ 个特征向量组成的矩阵进行标准化处理，最终组成 $n\times k_1$ 维的特征矩阵 $\boldsymbol{M}$。在实验过程中，为了能够快速地找出最优解，参数 $k_1$ 的选取基本等于聚类个数 k 或稍大于 k。

e)对特征矩阵 $\boldsymbol{M}$ 使用 k-means 算法进行最终得出被标记聚类类别的 n 个样本。

### 2.2.2 计算复杂度分析

本文算法第 1 步计算数值型数据的高斯核函数矩阵 $\boldsymbol{W}$，时间复杂度为 $O(mn^2)$，其中 n 为样本数目，m 表示数值属性维数；第 2 步计算分类型数据的影响因子矩阵 $\boldsymbol{F}$ 和构建相似度矩阵 $\boldsymbol{S}$，时间复杂度为 $O(2cn^2)+O(n^2)+O(2n)$，其中 n 为样本数目，c 表示分类属性维数；第 3 和 4 步对相似度矩阵 $\boldsymbol{S}$ 进行特征分解，时间复杂度为 $O(n^3)$；第 5 步对特征矩阵 M 进行 k-means 聚类，时间复杂度为 $O(knt)$，其中 k 表示聚类数目，t 表示 k-means 迭代次数。所以本算法的时间复杂度为 $O(n^3)+O((2c+m+1)n^2)+O((\mathrm{kt}+2)n)$。本文算法和传统的谱聚类算法在时间复杂度上基本处于同一个级别。

# 3 实验分析

为了研究 EBSCMD 算法的有效性，本文将其应用于 UCI 机器学习知识库中的混合数据集。实验中的操作系统为 Windows 10，集成开发环境为 Python 3。硬件条件为 CPU 为 Intel Core i7 2.8 GHz，内存为 8 GB。

## 3.1 实验结果分析

本节主要对 EBSCMD 算法进行对比实验与分析。从 UCI 机器学习知识库中选取了四个混合类型数据集，并与其他三个

经典算法进行了比较，验证本算法的可行性和有效性。

为了验证 EBSCMD 算法的有效性，实验中使用了从 UCI 机器学习知识库中获取的四个混合类型的数据集：Heart、Credit Approval 、Australian Credit Approval、Bank Marketing。这些数据集的详细信息由表 1 列出。表 1 中列举了四个数据集的聚类个数、维度（即数值属性个数加上分类属性个数）、和数据集中的对象个数。

表 1　混合类型数据集的详细信息

| Data set | Cluster | Dimension ( $M_n+M_c$ ) | N |
|---|---|---|---|
| Heart | 2 | 6 + 7 | 270 |
| Credit Approval | 2 | 6 + 9 | 653 |
| Australian Credit Approval | 2 | 6 + 8 | 690 |
| Bank Marketing | 2 | 7 + 9 | 4521 |

在这四个混合类型数据集的基础上，分别用 EBSCMD、K-Prototypes、OCIL、KL-FCM-GM 算法对以上数据集进行了聚类实验。同时，本文使用聚类精度（ACC）来度量聚类结果的准确度。对于 N 个不同的样本， $Y=(y_1,y_2,\cdots,y_n)$ 表示真正的类别标签， $C=(c_1,c_2,\cdots,c_n)$ 表示本文预测的聚类标签。ACC 的计算公式为

$$ACC=\sum_{i=1}^{N}\sigma\left(y_i,map(\mathrm{c}_i)\right)/\,\mathrm{N} \qquad (17)$$

这里 *map()*是通过匈牙利算法将每一个聚类标签映射到一个类别标签，这个映射是最优的，如果 $y_i=map(\mathrm{c}_i)$ 则 $\sigma\left(y_i,map(\mathrm{c}_i)\right)$ 就等于 1 或者 0。此外，N 是数据集中的对象个数，ACC 值越高，聚类的性能就越好。

EBSCMD 算法在实验中式(4)中的参数 σ 的变化是 1.0~15.0，参数值每次增加 0.5 来寻找最优的聚类效果。 K-Prototypes 算法中的参数 γ 的变化是 0.1~2.1，每次增加 0.1；KL-FCM-GM 算法中的参数 λ 的变化同样是 0.1~2.1，每次增加 0.1。在表 2 中本文列举出来了 EBSCMD、K-Prototypes、OCIL、KL-FCM-GM 算法四种算法的聚类精度。

EBSCMD、K-Prototypes、OCIL 和 KL-FCM-GM 算法在四个 UCI 数据集上的参数选择及聚类准确率分别在表 2~5 中列出。

从表 2 可以看出，算法 K-Prototypes、OCIL、KL-FCM-GM 在 Heart 数据集上的聚类准确率的均值分别为 0.783 0、0.741 1 和 0.792 6；而 EBSCMD 算法在 σ = 2.0 时聚类准确率为 0.833 3，比 K-Prototypes、OCIL、KL-FCM-GM 算法聚类准确率分别高出了 5.03%，9.22%，4.07%，因此 EBSCMD 算法性能更好。

表 2　Heart 数据集上的实验结果

| Data set | | Heart |
|---|---|---|
| EBSCMD | ACC | **0.8333** |
| | Para | k = 2，σ = 2.0 |
| K-Prototypes | ACC | 0.7830 ± 0.0445 |
| | Para | k = 2，γ = 0.2 |
| OCIL | ACC | 0.7411 ± 0.0678 |
| | Para | k = 2 |
| KL-FCM-GM | ACC | 0.7926 ± 0 |
| | Para | k = 2，λ = 0.8 |

从表 3 可以看出，算法 K-Prototypes、OCIL、KL-FCM-GM 在 Credit Approval 数据集上的聚类准确率的均值分别为 0.801 7、0.663 4 和 0.591 4；而 EBSCMD 算法在 σ = 13.0 时聚类准确率为 0.825 4，比 K-Prototypes、OCIL、KL-FCM-GM 算法聚类准确率分别高出了 2.37%、16.2%、23.4%，因此 EBSCMD 算法性能更好。

表 3　Credit Approval 数据集上的实验结果

| Data set | | Credit Approval |
|---|---|---|
| EBSCMD | ACC | **0.8254** |
| | Para | k = 2，σ = 13.0 |
| K-Prototypes | ACC | 0.8017 ± 0.0122 |
| | Para | k = 2，γ = 0.1 |
| OCIL | ACC | 0.6634 ± 0.0407 |
| | Para | k = 2 |
| KL-FCM-GM | ACC | 0.5914 ± 0.0847 |
| | Para | k = 2，λ = 0.3 |

从表 4 可以看出，算法 K-Prototypes、OCIL、KL-FCM-GM 在 Australian Credit Approval 数据集上的聚类准确率的均值分别为 0.795 5、0.666 8 和 0.831 9；而 EBSCMD 算法在 σ = 13.5 时聚类准确率为 0.831 9，比 K-Prototypes、OCIL、KL-FCM-GM 算法聚类准确率分别高出了 3.64%、16.51%、0.00%，因此 EBSCMD 算法性能更好。

表 4　Australian Credit Approval 数据集上的实验结果

| Data set | | Australian Credit Approval |
|---|---|---|
| EBSCMD | ACC | **0.8319** |
| | Para | k = 2，σ = 13.5 |
| K-Prototypes | ACC | 0.7955 ± 0.0180 |
| | Para | k = 2，γ = 1.0 |
| OCIL | ACC | 0.6668 ± 0.0382 |
| | Para | k = 2 |
| KL-FCM-GM | ACC | **0.8319 ± 0** |
| | Para | k = 2，λ = 1.5 |

从表 5 可以看出，算法 K-Prototypes、OCIL、KL-FCM-GM 在 Bank Marketing 数据集上的聚类准确率的均值分别为 0.613 4、0.624 5 和 0.540 0；而 EBSCMD 算法在 σ = 2.0 时聚类准确率为 0.635 0，比 K-Prototypes、OCIL、 KL-FCM-GM 算法聚类准确率分别低了 2.16%、1.05%、9.50%，因此 EBSCMD 算法性

能更好。

表 5 Bank Marketing 数据集上的实验结果

| Data set | | Bank Marketing |
|---|---|---|
| EBSCMD | ACC | **0.6350** |
| | Para | k = 2，σ =2.0 |
| K-Prototypes | ACC | 0.6134 ± 0.0817 |
| | Para | k = 2，γ = 0.2 |
| OCIL | ACC | 0.6245 ± 0.0372 |
| | Para | k = 2 |
| KL-FCM-GM | ACC | 0.5400 ± 0.0133 |
| | Para | k = 2，λ = 0.3 |

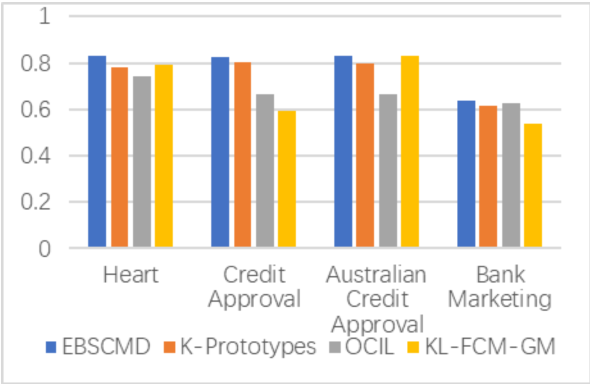



图 2 EBSCMD 与其他算法对比

图 2 汇总了本文算法和对比算法在四个 UCI 数据集上的聚类准确率。由图 2 可以看出，在所选取混合类型数据集上 EBSCMD 算法的聚类准确率是这四种算法中较高的，证明了 EBSCMD 算法的有效性，显示出本文算法在处理混合属性数据时的可行性。EBSCMD 算法具有较好的聚类效果的原因在于：EBSCMD 算法利用信息熵对混合数据中分类型数据进行处理，与利用高斯核函数计算数值型数据的相似度的方法相结合，避免了分类型数据和数值型数据的特征转换和参数调整。这种相似性度量方式应用简单，且具有广泛的覆盖性，并且 EBSCMD 算法在参数固定后运行稳定且不具有随机性，表明本文算法具有较高的稳定性以及良好的鲁棒性。

### 3.2 算法执行时间

表 6 列出了 EBSCMD 算法在四个 UCI 数据集上的平均执行时间。算法的执行时间和数据集的维度与数据量相关。从表 6 可以看出，Heart、Credit Approval、Australian Credit Approval 三个数据集的数据量相对较小，因此算法执行时间较短；而 Bank Marketing 数据集的数据量相对较大，因此算法执行时间较长。

表 6 EBSCMD 算法在各个数据集上的时间复杂度统计

| Data set | 平均执行时间/s |
|---|---|
| Heart | 2.6s |
| Credit Approval | 15.2s |
| Australian Credit Approval | 15.6s |
| Bank Marketing | 617.8s |

## 4 结束语

本文总结现有的混合型数据聚类算法原理以及各自的优缺点，提出了一种基于信息熵的混合数据谱聚类算法。该方法引入了一种新的基于信息熵的混合型数据的相似性度量用于谱聚类算法中，基于信息熵的方法处理混合型数据中的分类型数据，建立起分类型数据间的关联，并与数值型数据的高斯核函数计算方式相结合，更准确地计算了混合类型数据的相似性度量；又把这种相似性度量方式与谱聚类算法相结合，使算法可以处理任意形状的数据，从而提高了聚类准确度以及算法的鲁棒性。

EBSCMD 算法能有效地处理混合属性数据的聚类问题。虽然建立了分类型数据间的关联，但并没有考虑数值型数据间的关联以及其本身重要度对实验结果的影响。在下一步的研究工作中，将考虑摆脱谱聚类算法本身的高斯核函数矩阵，用信息熵的方式处理数值型数据，建立其内部的关联性，进一步的对混合属性数据进行高效地聚类。

## 参考文献：

[1] On N I, Boongoen T, Kongkotchawan N. A new link-based method to ensemble clustering and cancer microarray data analysis [J]. International Journal of Collaborative Intelligence, 2014, 1 (1): 45.

[2] Ludwig S A. MapReduce-based fuzzy C-means clustering algorithm: implementation and scalability [J]. International Journal of Machine Learning & Cybernetics, 2015, 6 (6): 923-934.

[3] Bouras C, Tsogkas V. Assisting cluster coherency via n-grams and clustering as a tool to deal with the new user problem [J]. International Journal of Machine Learning & Cybernetics, 2014, 7 (2): 1-14.

[4] Lloyd S. Least squares quantization in PCM [J]. IEEE Trans on Information Theory, 1982, 28 (2): 129-137.

[5] 张涛，刘昶，周晓锋，等. 基于真实核心点的密度聚类方法 [J//OL]. 计算机应用研究, 2018, 35 (12): 1-7. (Zhang Tao, Liu Chang, Zhou Xiaofeng, *et al*. Density clustering algorithm based on real core point [J//OL]. Application Research of Computers, 2018, 35 (12): 1-7. )

[6] 王日宏，崔兴梅，李永珺. 自适应调整的布谷鸟搜索 K-均值聚类算法 [J//OL]. 计算机应用研究, 2018, 35 (12): 1-7. (Wang Rihong, Cui Xingmei, Li Yongjun. Self-adaptive adjustment of cuckoo search K-means clustering algorithm [J//OL]. Application Research of Computers, 2018, 35 (12): 1-7. )

[7] 张文倩，庄华亮，陈翔，等. 基于竞争思想的分级聚类算法 [J]. 信息与控制, 2017, 46 (5): 614-619. (Zhang Wenqian, Zhuang Hualiang, Chen Xiang, *et al*. Hierarchical clustering algorithm based on competitive learning [J]. Information and Control, 2017, 46 (5): 614-619. )

[8] Huang Zhexue. A fast clustering algorithm to cluster very large categorical data sets in data mining [C]// Proc of Research Issues on Data Mining & Knowledge Discovery, 1997: 1-8.

[9] Daniel Barbará, Li Y, Couto J. COOLCAT: an entropy-based algorithm for categorical clustering [C]// Proc of DBLP. 2002: 582-589.

[10] 李桃迎，陈燕，张金松，等. 一种面向分类属性数据的聚类融合算法研究 [J]. 计算机应用研究, 2011, 28 (5): 1671-1673. (Li Taoying, Chen Yan, Zhang Jinsong, *et al.* Clustering ensemble algorithm for categorical data [J]. Application Research of Computers, 2011, 28 (5): 1671-1673. )

[11] 陈晋音，何辉豪. 基于密度和混合距离度量方法的混合属性数据聚类研究 [J]. 控制理论与应用, 2015, 32 (8): 993-1002. (Chen Jinyin, He Huihao. Density-based clustering algorithm for numerical and categorical data with mixed distance methods [J]. Control Theory and Application, 2015, 32 (8): 993-1002. )

[12] Li Cen, Biswas G. Unsupervised learning with mixed numeric and nominal data [J]. IEEE Trans on Knowledge & Data Engineering, 2002, 14 (4): 673-690.

[13] Huang Zhexue. Extensions to the K-means algorithm for clustering large data sets with categorical values [J]. Data Mining & Knowledge Discovery, 1998, 2 (3): 283-304.

[14] Chatzis S P. A fuzzy c-means-type algorithm for clustering of data with mixed numeric and categorical attributes employing a probabilistic dissimilarity functional [J]. Expert Systems with Applications, 2011, 38 (7): 8684-8689.

[15] Cheung Y M, Jia H. A unified metric for categorical and numerical attributes in data clustering [M]// Advances in Knowledge Discovery and Data Mining. Berlin: Springer, 2013: 135-146.

[16] Ding Shifei, Du Mingjing, Sun Tongfeng, *et al.* An entropy-based density peaks clustering algorithm for mixed type data employing fuzzy neighborhood [J]. Knowledge-Based Systems, 2017 (133): 294-313.

[17] Fan R K C. Spectral graph theory [M]// American Mathematical Society. 1997.

[18] 乔晓明，潘晓英. 基于稀疏图的鲁棒谱聚类算法 [J//OL]. 计算机应用研究, 2018, 35 (6): 1-2. (Qiao Xiaoming, Pan Xiaoying. Robust spectral clustering based on sparse graph [J//OL]. Application Research of Computers, 2018, 35 (6): 1-2. )

[19] 马恒，丁世飞. 一种基于混合数据相似性度量的谱聚类算法 [J]. 小型微型计算机系统, 2016, 37 (8): 1746-1750. (Ma Heng, Ding Shifei. Spectral clustering algorithm based on of mixed data similarity measure [J]. Journal of Chinese Computer System, 2016, 37 (8): 1746-1750. )